https://doi.org/10.17576/apjitm-2018-0702-08

http://www.ftsm.ukm.my/apjitm
Asia-Pacific Journal of Information Technology and Multimedia
Jurnal Teknologi Maklumat dan Multimedia Asia-Pasifik
Vol. 7 No. 2, December 2018: 99 - 113
e-ISSN: 2289-2192

AN ENSEMBLE FEATURE SELECTION METHOD TO DETECT WEB SPAM

MAHDIEH DANANDEH OSKOUEI
SEYED NASER RAZAVI

ABSTRACT

Feature selection is an important issue in data mining, and it is used to reduce dimensions of features set. Web spam detection is one of research fields of data mining. With regard to increasing available information in virtual space and the need of users to search, the role of search engines and used algorithms are important in terms of ranking. Web spam is an illegal method to increase mendacious rank of internet pages by deceiving the algorithms of search engines, so it is essential to use an efficient method. Up to now, many methods have been proposed to face with web spam. An ensemble feature selection method has been proposed in this paper to detect web spam. Content features of standard dataset of WEBSPAM-UK2007 are used for evaluation. Bayes network classifier is used along with 70-30% training-testing spilt of dataset. The presented results show that Area Under the ROC Curve (AUC) of this method is higher than the other methods reported in this paper. Moreover, the best values of evaluation metrics in our proposed method are optimal in comparison to the other methods reported in this paper. In addition, it improves classification metrics in comparison to basic feature selection methods.

Keywords: *Ensemble feature selection, Web spam, Ranking, Machine learning.*

INTRODUCTION

Internet is a global information system. Most of users use search engines due to high volume of information in virtual world in order to access required information. They often observe the results of the first pages in search engines. If they cannot obtain desired results, then they exchange query statement. Search engines try to place the best results in the first links of results on the basis of user's query. Web spam is considered as an intruder in search engines. The purpose of creating them is to penetrate ranking algorithms of search engine. Therefore, search engine tries to struggle with this problem. In recent years, search engine optimization (SEO), by using legal methods, help the websites reach the higher rank in various search engines. This method is time consuming and costly. In contrast, another method is to use web spamming to increase the rank of search engines. It not only decreases the quality of search engines and the trust between the users and providers of search engines but also wastes computing resources of search engines. Therefore, a competition exists between spammers to achieve the high rank in search engines and managers of search engines to present related valid results.

One of detection methods of web spam is to use machine learning methods. Valid and spam pages have different statistical features. These differences can be used to create automatic classification. In machine learning methods, the classifier predicts that whether web page or web site is a spam or not, and this prediction is performed on the basis of web pages features.

Feature selection is an important pre-processing step helping to increase the efficiency of prediction in a model. Feature selection involves two methods of feature ranking and selection of feature subset. In this paper, we present a new ensemble method for feature selection. In order to create an ensemble list, we used features selected by two techniques involving feature ranking and selection of feature subset. In our method, the ensemble list is created by applying the considered threshold on frequency and F-score value of each feature in selected features lists. The presented method is called EFS-FF (ensemble feature selection based on frequency and F-score). In addition, we used nine various feature selection methods in our experiments. Among these techniques, 2 methods are related to feature subset selection approaches, and 6 methods are ranking feature selection approaches. We presented results of 16 different ensemble based methods in total. Also, we used Bayes network for classification. The results show that the proposed method demonstrate higher results than when feature selection is not included in the classification. Also, the proposed method demonstrate the higher results than when single feature selection is included in the classification. To estimate the effectiveness of our method, we compared our method with basic feature selection methods and the results reported from the methods of web spam detection with the same dataset. The results show that the method of ensemble feature selection presented in this paper involves the higher results, and it improves spam pages classification.

The rest of this paper is organized as follows. In section II, we present related studies in terms of web spam detection. In section III, we review basic feature selection methods. In section IV, we propose the framework of our proposed method. In section V, we describe the results of evaluation, and finally, in section VI, we present conclusions and future work.

LITERATURE REVIEW

Spam page are defined as an activity performed by human intentionally so that the location of internet page is changed (Gyongyi & Garcia-Molina, 2005). In another definition, these pages are introduced as web pages involving hyperlink to mislead search engines (Boldi, 2005). Researchers have presented various web spam detection methods. One of detection methods is natural language processing (NLP). Westbrook & Greene (2002) used semantic analysis of text content to detect web spam. Cafarella & Cutting (2004) suggested that if more phrases are continuously displayed, then search engines remove and delete repetitive phrases. There are a number of link-based methods, and we refer to some of them. Algorithms such as Page Rank and HITS algorithms are taken into account to struggle with spam. The graph-based approach was used to detect link farms. B & B.D2006) (used two-part sub graphs to detect farms. Li et al.(2002) carried out the research in terms of improving HITS efficiency. They showed that the pages having less input links and more output links had worse HITS results. Eiron et al.(2004) showed that Host Rank was resistant against link spam. Ng et al.(2001) analyzed HITS and PageRank algorithms, and proposed two improved algorithms of HITS involving random HITS and virtual space HITS. Becchetti et al.(2006) suggested using Truncated PageRank algorithm to struggle with link-based spam. Acharya et al.(2008) considered historical data to detect spam pages. They stated that heterogeneous growth rage in return links was an indication of spam. If a page was obtained for incompatible set from queries, then it was probably a spam. Chakrabarti et al.(2001) proposed using DOM tree. They detected the tree of document object model (DOM) for structure pages and sub-tree corresponding with other parts. Then, such a sub-tree showed a special behavior in mutual reinforcement process. Zhang & Li (2006) used content quality and link quality based on a distribution method based on trust to struggle with web spam.

Researchers used various machine learning methods to detect web spam. Davison (2000) used machine learning methods to detect Nepotistic links. Also, machine learning methods such as SVM were used to detect spam blogs (Kolari, Finin, & Joshi, 2006; Kolari, Java, Finin, Oates, & Joshi, 2006). Prieto et al.(2012) presented a system called SAAD in which web content was used to detect web spam. In this method, C4.5, Boosting and Bagging were used for classification. Amitay et al.(2003) used classification algorithms to detect the capabilities of a website, and detected 31 clusters. Each one was considered as a group of spam. Ntoulas et al.(2006) used pages content features to detect spam pages. The results showed that machine learning was a promising method to struggle with content-based spam. Danandeh Oskouei & Razavi (2015) used danger theory to detect web spam. Their method was based on machine learning. Rungsawang et al.(2011) considered ant colony optimization algorithm to classify web spam. The obtained results showed that this method had higher precision and lower Fall-out in comparison to SVM and decision Tree. Silva et al.(2012) investigated various classification methods such as decision tree, SVM, KNN, adaBoost, Bagging and LogitBoost to detect web spam on different features. Karimpour et al.(2012) reduced the number of features by using PCA, and then they considered semi-supervised classification method of EM-Naive Bayesian to detect web spam. Fdez-Glez et al.(2015) presented a filter method of web spam called WSF2 that was a quick learning along with increasing learning to classify web spam. It was designed by using CBR method. Jayanthi & Sasikala(2011) proposed a method called DBLCSPAMCLUST to detect web spam . They used k-mean clustering in their method. Also, in an other paper, they used fuzzy c-mean clustering to detect link-based web spam (Jayanthi & Sasikala, 2010). Jayanthi & Sasikala(2013) proposed a method based on Reptree (Regression tree representative) to detect web spam. Link-based features were used to detect web spam in this study.

Researches used genetic algorithms in identifying web spam, and we refer to two studies. Jayanthi & Sasikala (2012) presented a method based on genetic algorithm to detect spams involving link, farm and clique spam. In another study, Sasikala(2012) classified link-based spam by using two methods involving GA Decision tree and J48 Decision tree. The results showed that genetic-based classification method has higher accuracy

In this paper, we presented the ensemble method of feature selection to improve web spam classification. In experiments, the highest value of AUC was better than AUC reported values by web spam challenge Workshop (2008) and reported AUC by Fdez-Glez et al. (2015). In addition, our best results were optimal in comparison to basic feature selection methods and results reported by Keyhanipour & Moshiri (2013).

FEATURE SELECTION

Feature selection is an important pre-processing step in data mining, and it is a technique to select the best features subset to create an optimized learning model by using some evaluation criteria. The methods of feature selection can be classified to three group involving filter, wrapper and embedded methodes (Bellotti, Luo, & Gammerman, 2006; Guyon & Elisseeff, 2003). In addition, feature selection techniques can be categorized into two types involving feature subset selection and feature ranking (Liu, Motoda, Setiono, & Zhao, 2010). In feature subset selection methods, subsets of attributes are selected in a way that they collectively have good predictive capability. In feature ranking, attributes are evaluated individually. Also, rank of each attribute is evaluated according to its individual predictive capability (Gao, Khoshgoftaar, & Napolitano, 2014). The filter methods are used as feature ranking strategies and the wrapper and the embedded methods are used as feature subset selection strategies (De Silva & Leong, 2015). In the filter method, the features are selected on the basis of pre-processing step in which learning algorithm is ignored. This method is created on the basis of inherent features rather than a special classifier. In this method, features are scored on the basis of some criteria. In this way, a score is dedicated to each feature, and then the scores are ordered. Afterwards, k numbers of best features are selected. Finally, this set is classified by using a classifier. In the wrapper method, the features set is selected on the basis of classification method, and search methods like SFS and SBS are used. In this approach, all subsets of features are taken into account. Through evaluating all modes, the best one having minimum error is selected. In the embedded method, the advantages of two previous approaches are used by using different evaluation criteria.

The feature selection methods used in experiments are reviewed below.
Chi-squared is a method of feature ranking based on filter, and it is on the basis of x^2-statistic (Liu & Setiono, 1995). In this method, features are independently evaluated on the basis of class labels. Chi-square of each feature is computed by using formula 1 in this method:

$$X^2=\sum_{i=1}^{I}\sum_{j=1}^{B}\frac{\left[A_{ij}-\frac{R_i*B_j}{N}\right]^2}{\frac{R_i*B_j}{N}} \text{ . (1)}$$

I is the number of distances. B is the number of classes, N refers to the number of samples, R_i stands for the number of samples in Ith distances, B_j is the number of samples in Ith class and A_{ij} is the number of samples in the Ith distance and Ith class.
Gain Ration (GR) is a method of feature ranking based on filter. GR maximizes information gain of features, and minimizes their values number. Gain ration of x is obtained by dividing IG of x to inherent value (Hall & Smith, 1998).

$$GR(x)=IG(x)/IV(x) \text{ . (2)}$$

Inherent value of x feature is defined by formula 3:

$$IV(X)=-\sum_{i=1}^{r}\left({|X_i|}/{N}\right)\log\left({|X_i|}/{N}\right). \quad (3)$$

$|x_i|$ is the number of samples in which that feature receives x_i value. r is distinct number of x, and N stands for whole number of samples in dataset.
IG is a method of feature ranking based on filter. Information Gain IG (x/y) assesses the importance of a feature based on the amount by which the entropy of x decreases the values of y (Hall & Smith, 1998). IG(X/Y) is calculated by formula 4:

$$IG(X|Y) = H(X) - H(X|Y) \text{ . (4)}$$

H(X) is calculated by the formula 5:

$$H(X)= -\sum_{x\in X} p(x)\log p(x) \quad (5)$$

. H(X|Y) is computed by formula 6:

$$H(X|Y)= \sum_{y\in Y} p(y)\sum_{x\in X} p(x|y)\log p(x|y) \qquad (6)$$

ReliefF is a method of feature selection based on filter, and it is an expansion of Relief. It is inherently used to solve two-class problems. It can be used for multi-classes problems by dividing the problem into series of two-class problems (Kononenko, 1994). In this method, dataset is randomly sampled, and the value of a feature is evaluated by repeating the sampling according to the feature value of the nearest neighbor with similar and different classes.

OneR is a rule-based algorithm (Holte, 1993). The method use for evaluating the feature based on the wrapper ranks the feature by using OneR classification method and on the basis of error rate in ranking training set. It creates simple rules on the basis of a feature. It creates the rules at the same time, and tests the unit feature. A branch is created for the value of feature.

CFS is a filter-based algorithm by using a search algorithm along with a function to calculate competency of feature subsets (Hall, 1998). In this algorithm, a feature subset is selected according to correlation-based heuristic evaluation. The basis of evaluation function is subsets involving features that have higher relation with the class, and they are not related to

each other. Since unrelated features have lower relation with the class, they are ignored. In order to reduce computation cost, search algorithm is used. In this paper, we used Best First to search. It searches the space of feature subsets by using greedy fill climbing completed by backtracking facility.
Consistency is a feature selection method based on filter using a search algorithm with a function (Liu & Setiono, 1996). In this method, the value of a set of features is calculated by class values when training samples are projected on a subset. If produced subset contains lesser features than the best recent subset, then inconsistency is compared with inconsistency index of the best subset. If it is more consistent than the best subset, then the produced subset is considered as the best subset. In this paper, we use genetic algorithm to search.

F-score is a feature selection method based on filter (Chen & Lin, 2006). The value of F-score of each feature is calculated by formula 7. Features with larger values of F(i) are more discriminative. In this paper, we considered F-score average of all features as threshold value. Therefore, if F-score of each feature is larger than threshold value, then that feature is selected.

$$F(i)=\frac{\left(\bar{x}_i^{(+)}-\bar{x}_i\right)^2+\left(\bar{x}_i^{(-)}-\bar{x}_i\right)^2}{\frac{1}{n_+-1}\sum_{k=1}^{n_+}\left(x_{k,i}^{(+)}-\bar{x}_i^{(+)}\right)^2+\frac{1}{n_--1}\sum_{k=1}^{n_-}\left(\bar{x}_{k,i}^{(-)}-\bar{x}_i^{(-)}\right)^2}\ .\quad (7)$$

n_+ and n_- are respectively the number of positive and negative samples. $\bar{x}_i$, $\bar{x}_i^{(+)}$ and $\bar{x}_i^{(-)}$ are respectively ith average of the feature, ith average of positive feature, and ith average of negative feature. $X_{k,i}^{(+)}$ and $X_{k,i}^{(-)}$ are respectively ith average of the feature from kth of positive sample and ith feature of kth negative sample (Chen & Lin, 2006).

THE MECHANISM OF ENSEMBLE FEATURE SELECTION

Feature selection methods are applied to classification problems by choosing a reduced subset feature of the basic set to achieve the faster and more accurate classification. Feature and model selection are two important factors in creating a desirable classification (Koc, Mazzuchi, & Sarkani, 2012). There are many methods to select the appropriate feature. Studies on feature selection methods show that using a combination of feature selection methods can improve the performance of classifications by identifying features that are weak as individuals but strong as a group and by eliminating redundant features and determining features that have high correlation with output class (Bolon-Canedo, Sanchez-Marono, & Alonso-Betanzos, 2011; Wang, He, Liu, & Gombault, 2015).
In ensemble feature selection, a design similar to ensemble classification is used. Ensemble feature selection method involves two steps. At first, A set of various ranking lists is created by using rankers. In the next step, these ranking lists are integrated by using rank aggregation of the features.
Ensemble feature selection reduces the variability resulting from use of a single feature selection method. (Dittman, Khoshgoftaar, Wald, & Napolitano, 2013) Ensemble feature selection methods can be divided into two groups.
Homogeneous distributed ensemble: in this method, ensemble of a single feature ranking technique is created using a feature selection method and different training data, and then the final list of the selected features is obtained using a combination method (Seijo-Pardo, Porto-Díaz, Bolón-Canedo, & Alonso-Betanzos, 2017).
Heterogeneous centralized ensemble: in this method, ensemble of multiple feature ranking technique is generated by different feature selection method and the same training data. The feature lists are combined by a combination method to obtain the final list of the features(Seijo-Pardo et al., 2017).

The ensemble technique has more accurate and stable results due to the use of different feature selection methods. These methods evaluate the important and different qualities of the data, so that combining these methods leads to an optimal performance in camparison to individual methods (Dahiya, Handa, & Singh, 2016).The key step in ensemble feature selection is how to aggregate the results. There are various methods to select features having advantages and disadvantages. These methods including mean, median, the highest rank, the lowest rank and etc.Most ensemble methods are based on creating multiple ranking lists and then aggregating them to the final ranking for each feature. There is also another method that aggregates scores from the selected metric rather than ranking based on scores (Dittman et al., 2013).

Some studies are presented in the field of ensemble feature selection (Osanaiye et al., 2016) presented EMFFS method for DDoS detection in cloud computing. It is an ensemble feature selection method that includes four filter-based feature selection methods. In this method, one third of the output of each filter-based method is selected, and if the number of each feature derived from four methods is greater than threshold, then that feature is selected. Hoque, Singh, & Bhattacharyya (2018) Presented an ensemble method. It is in fact an ensemble feature selection method including five filter based methods. In this method, the number of final features selected is max *k*, and if a feature is selected using all five based methods, it is considered as the selected feature. Otherwise, Mutual Information is used to determine whether or not to select a feature. Silwattananusarn, Kanarkard, and Tuamsuk (2016) used the ensemble machine learning and ensemble feature selection to classify the Cardiotocogram data. The results of the experiments obtained from the proposed method showed that the accuracy of the classification increased. Sahu, Dehuri, and Jagadev (2017) used ensemble selection features in pipeline with GA coupled by multi-objective optimization to increase the accuracy of prostate cancer data. The results showed that the proposed method compared with Group Genetic Algorithm (GGA) had stable results, and also, it was more effective in the selection of relevant features.

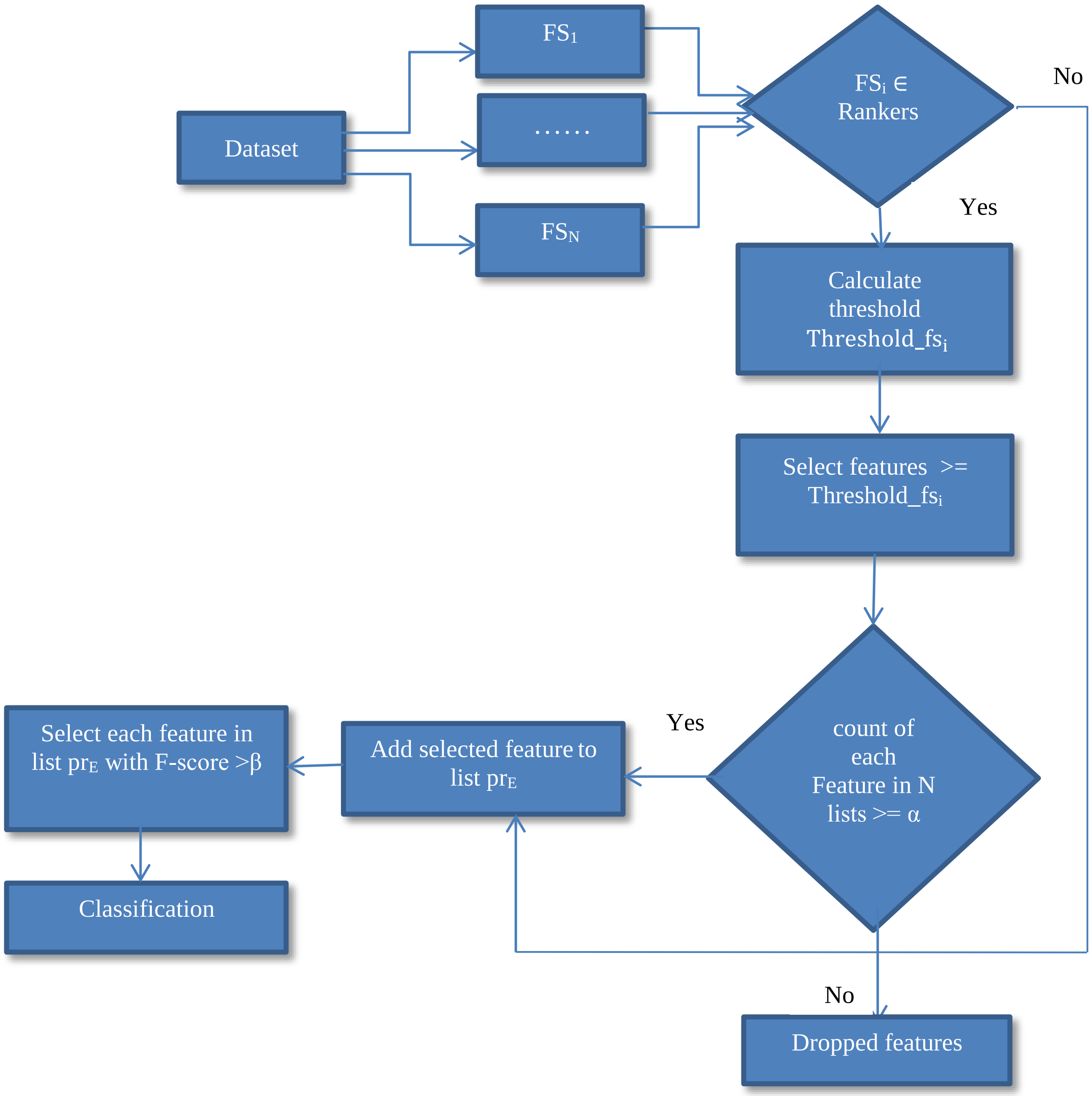

FIGURE 1. Process of the proposed method.

EFS-FF TECHNIQUE

Our proposed method (EFS-FF) is an ensemble approach that combines the subsets obtained from feature ranking and feature subset selection methods as mentioned in Figure 1. Also, if feature subset selection method is used, then selected features list are used without any changes in final decision making. But, if the feature ranking method is used, then after computing the score of each feature, we apply threshold on this list, and we delete unimportant features. To decide on the final list, the importance of each feature is determined according to the number of repetitions of each feature in features lists obtained by different feature selection methods and F-score value of each feature. Hence, the final list is obtained by applying considered threshold on the frequency and f-score of each feature.
The process of the proposed method is shown in Figure 1, and details of the proposed method are as follows:

We assume that D is the dataset involving R samples and W features, $D=X_1,X_2,\ldots,X_R$. Also, we suppose that E is the ensemble involving N methods of feature selection (FS), $E=\{Fs_1,Fs_2,\ldots,Fs_N\}$. N lists of selected features are obtained by using feature selection methods. In feature ranking methods, we considered the score of each feature as selection or deletion criterion. Also, we considered the average score of features as threshold value in each list. Threshold_fs$_i$=AVG $score_i^j$, ($j = 1, \ldots, W$).Threshold_fs$_i$ is threshold value in ith list and $score_i^j$ is the score of jth feature computed by FS_i. If Fs_i is the method based on ranking, then the features are selected by applying the threshold. If the score of each feature is larger than or equal to Threshold_fs$_i$, then that feature will be selected. If feature subset selection is used, then selected features list is used without applying this threshold in final decision making. In terms of feature subset selectors of CFS and Consistency, the Best First and genetic algorithm are used to search respectively.

In next step, combination method is used to obtain final subset. Its parameters are frequency and F-score value of features. Frequency of each feature is obtained by counting the number of that feature in all N lists. If frequency of a feature is greater than or equal to N-1, Then, that feature is added to list pr_E, and F-score value of feature is determinant in its selection or unselection, list pr_E is primary ensemble list.

In next step, if F-score value of feature in list pr_E is larger than β, β is average F-score value of all features, then that feature is added to the final list. The steps of the algorithm are shown in Algorithm 1.

Algorithm 1: EFS-FF

Input:

1. Data set D with R instances and W features. F_j ,($j = 1, \ldots, W$);
2. $score_i^j$ is the score of jth feature obtained by Feature ranker Fs_i;
3. ensemble E with N method of feature selection (FS), $E=\{Fs_1,Fs_2,\ldots,Fs_N\}$;
4. Each instance $R \in D$ is assigned to one of two classes;
5. $_{pre}$defined thresholds:
    a. Threshold_fs$_i$: Threshold value on score of features in ith list to be selected.
    b. Frequency α: threshold value on frequency of features to be selected.
    c. F-score value β: threshold value on F-score of features to be selected.

Output:

Selected feature subsets.

Steps:

Apply S feature ranking method to dataset D;
for feature ranker Fs_i, i = 1, . . . , s do
    Calculate threshold Threshold_fs$_i$ using averag score of features in Fs_i.
    Select features with score larger than or equal to Threshold_fs$_i$ ($score_i^j$ >= Threshold_fs$_i$).
Select Best First and genetic algorithm to search for CFS and Consistency respectively.
Apply CFS and Consistency methods to dataset D;
Select all selected features using CFS and Consistency methods.
Calculate F-score value of all features.
β = average F-score value of all features.
α = N-1
For F_i in N lists
    If count of F_i in N lists >= α
        Select F_i and add to list pr_E
For F_i in list pr_E
    Select each feature in list pr_E with F-score value larger than β and add to final list

RESULTS EVALUATION

In this paper, we presented a novel ensemble feature selection method. In our method, ensemble list is constructed based on thresholds of frequency and F-score of features. In order to show the advantage of proposed ensemble method, it is applied to WEBSPAM-UK2007 dataset. We compared results of proposed method with variant used basic feature selection methods and some web spam detection methods. We used Bayes Net classifier for classification. The dataset is randomly splitted into 70% for training and 30% for testing. Feature selection methods used in experiments are chi-squared, Gain Ratio, IG, ReliefF, OneR, CFS, Consistency and F-score.

DATASET

In our experiments, we used dataset of WEBSPAM-UK2007 to compute evaluation metrics. This is a publicly available dataset used in web spam, and is labeled by a group of volunteers collected from UK in May 2007. WEBSPAM-UK2007 includes 105.9 million pages and over 3.7 billion links for about 114,529 hosts. In this dataset, there are three categories of features that are as follows:

1. Obvious features that include two attributes, number of attributes, and number of pages.
2. Content-based features that were extracted from the content of web pages and include features such as number of words in the page, number of words in the title, average word length, and so on.
3. Link-based features that were extracted from the link structure between web pages and include features such as in-degree, out-degree, PageRank, TrustRank, Truncated PageRank, and so on.

We used content features in experiments, and it contains 96 features. Features employed in this paper are listed in Appendix II.

EVALUATION METRICS

We used the following metrics in order to evaluate the performance of proposed algorithm: *Precision, Accuracy, F-Measure, AUC* and *FP Rate*.

Precision: it is the proportion of sample numbers that are truly detected as spam pages to the total number of samples that are detected as positive.

$$Precision = \frac{\text{True positive}}{\text{\# Predicted Positive}} = \frac{\text{True positive}}{\text{True Positive + False Positive}}. \quad (8)$$

Accuracy: Accuracy refers to the proportion of samples accurately classified to total number of samples.

$$Accuracy = \frac{\text{True Positive+True Negative}}{\text{True Positive + False Positive + True Negative + False Negative}}. \quad (9)$$

F-Measure: It is a harmonic mean of precision and recall.

$$\text{F-Measure}=2\times\frac{\text{Precision*Recall}}{\text{Precision+recall}}. \quad (10)$$

AUC: The AUC represents the area under the ROC curve. AUC is a statistically consistent and more discriminating measure than accuracy. The higher AUC is better and shows that the classifier has the higher true positive rate. The ROC curve is a method for checking the performance of the classifiers. In fact, ROC curves are two-dimensional curves in which the

True Positive Rate (TPR) is plotted on the Y axis and similarly False Positive Rate (FPR) is plotted on the X axis. In other words, a ROC curve shows the relative compromise between profits and costs.

COMPARING RESULTS OF PROPOSED METHOD WITH OTHER METHODS

In this part, performance of proposed algorithm is compared with the results classification of all features and the results of basic feature selection methods and some detection web spam methods. Tables are listed in Appendix I.Table 1 presents the results obtained from basic feature selection methods. Table 2 presents the obtained results of various combinations in novel proposed algorithm. As it is observed in table 2, the results of several 2-5 groups of feature selection methods are presented in proposed algorithm. Combination i, j, ..., n in the text and table 2 indicates the use of the feature selection methods with numbers i, j, .., n presented in Table 1 to create an ensemble feature list.

By studying the results of Table 2, it can be observed that combination number 11 involves the best values in metrics of Precision and Accuracy. In this combination, The number of features are reduced from 96 to 10. Also, combination number 14 in evaluation metrics of AUC and F-measure has the highest values in comparison to other results of this table, all features and feature selection methods in table 1. In this combination, The number of features are reduced from 96 to 15. In Figure 2 to Figure 4, the number of the results presented in Table 2 are compared with the results presented in table 1. As it is observed in each Figure, 5 combinations involving the better values in the proposed method have optimal values in comparison to the results of using all features and the results of basic methods of feature selection presented in table 1. In Figure 2, the precision of the proposed method is compared with the basic feature selection methods. In this Figure, five different combinations in the proposed method are compared with the basic methods. As it can be seen in figure, the combination 1, 2, 3 and 4, using the methods of feature selection of numbers 1, 2, 3 and 4 in Table 1, has the precision value of 0.328 that is the best value. In Figure 3, F-measure metric of the proposed method is compared with the basic feature selection methods. In this figure, five different combinations in the proposed method are compared with the basic methods. As it can be observed, the combination of 2, 3, 4, 6 in the proposed method has the F-measure value of 0.359, considered as the highest value. Figure 4 compares the accuracy of the proposed method with the basic feature selection methods. In this figure, five different combinations in the proposed method are compared with the basic methods. The combination of 1, 2, 3, 4 in the proposed method has the accuracy value of 0.93, that is the best value.

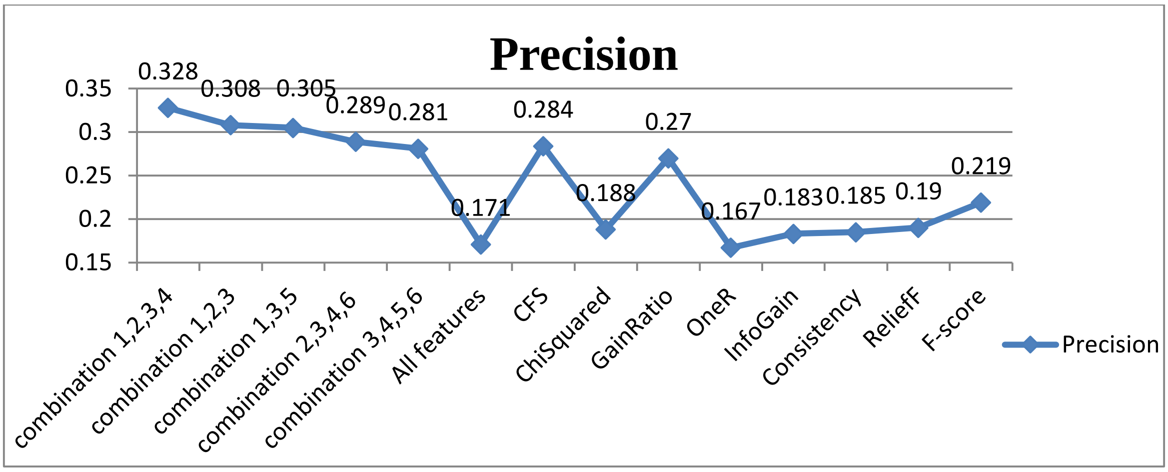

FIGURE 2. Comparing Precision of the higher results in the proposed method and basic feature selection methods.

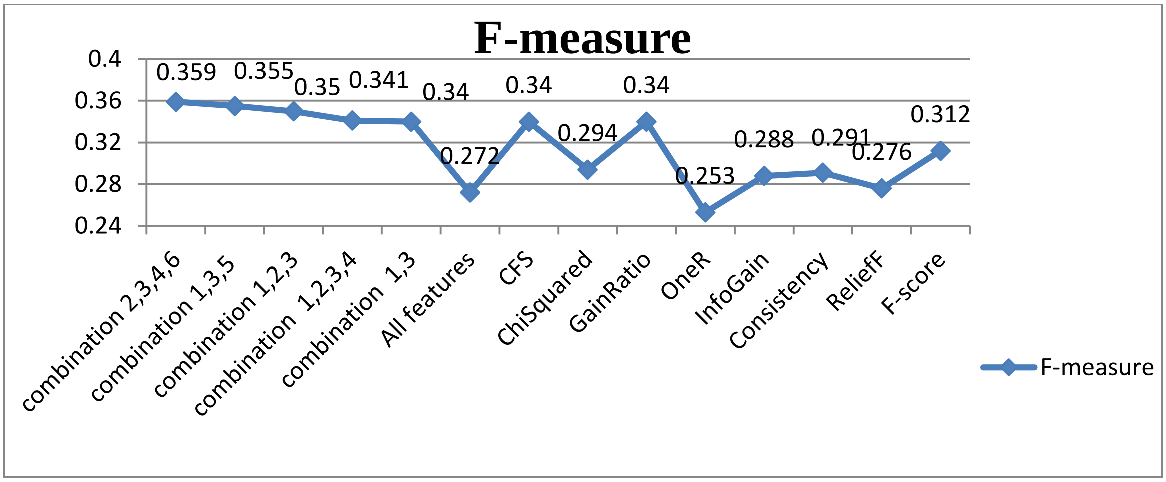

FIGURE 3. Comparing F-measure of the higher results in the proposed method and basic feature selection methods.

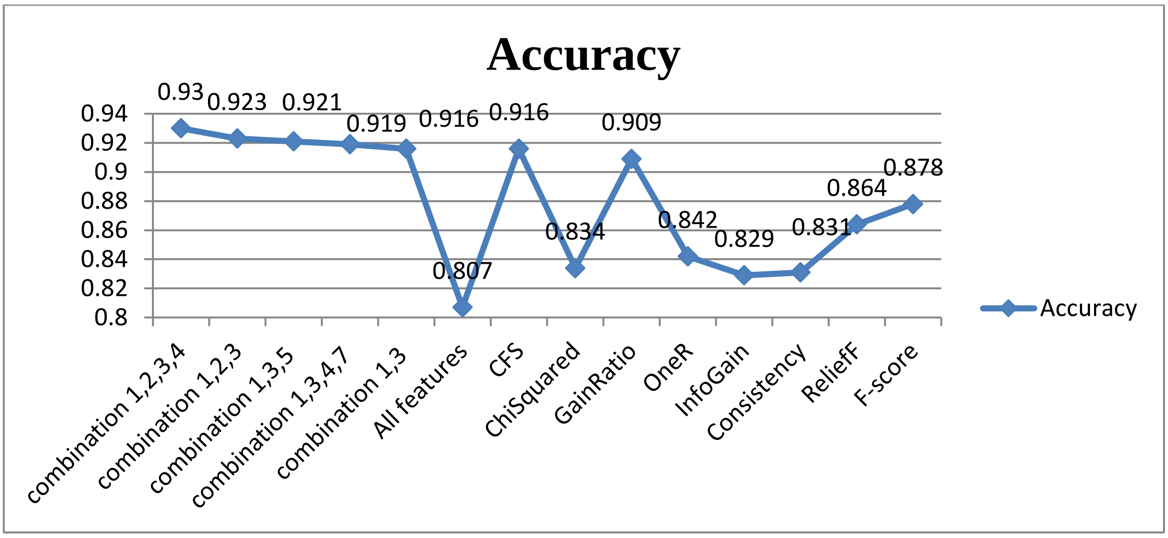

FIGURE 4. Comparing accuracy of higher results in the proposed method and basic feature selection methods.

In order to be sure of performance in the proposed method, the highest value of AUC metric in table 2 is compared with reported AUC values for top-ranked participant teams results of web spam challenge Workshop (2008) and AUC value reported by Fdez-Glez et al.(2015) in figure 5. As it can be observed, the best value of AUC in our proposed method is 0.851, and it is higher than results of web spam challenge Workshop (Workshop, 2008) and (Fdez-Glez et al., 2015). Also, AUC values of all combinations in our experiments is higher than AUC of reported by Fdez-Glez et al.(2015). Moreover, the best results in the proposed algorithm are compared with the results reported by (2013) in figure 6. Moreover, the best values of evaluation metrics of AUC, Precision, Accuracy and F-measure in our proposed method are optimal in comparison to the values reported by Keyhanipour & Moshiri (2013).

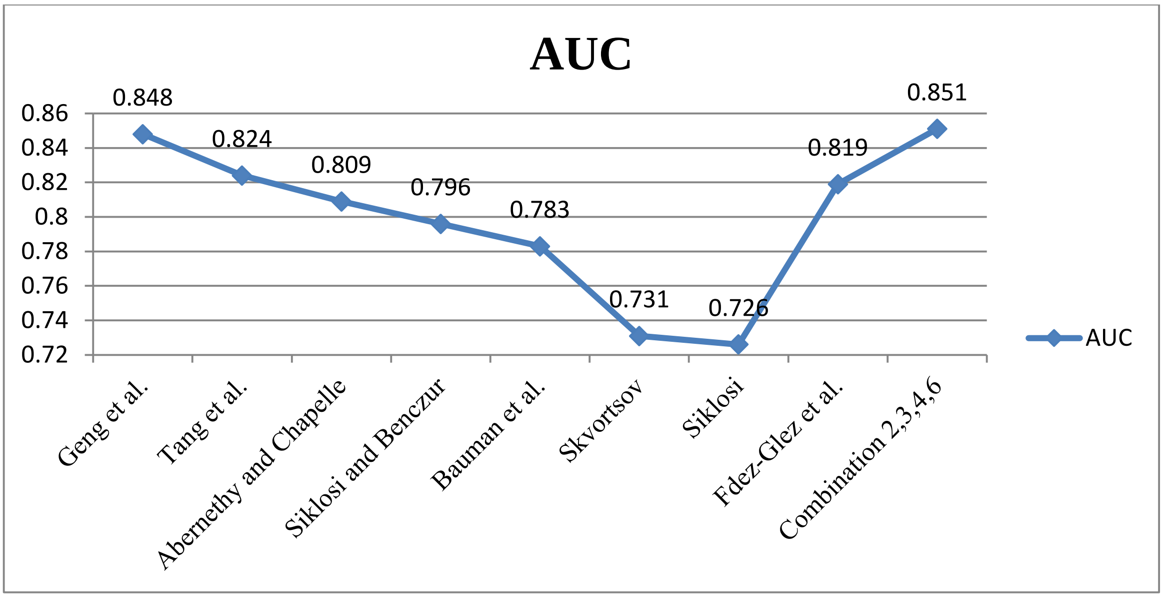

FIGURE 5. Comparing the highest metric of AUC in the proposed method with the results of web spam challenge (2008) and AUC of reported method (Fdez-Glez et al., 2015).

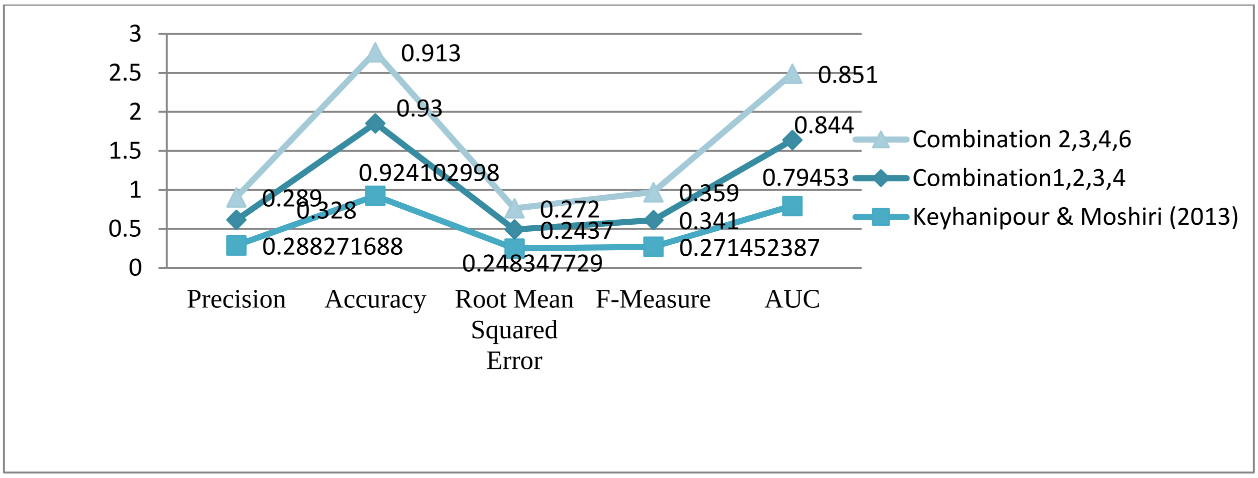

FIGURE 6. Comparing evaluation metrics with the results reported by Keyhanipour & Moshiri(2013).

CONCLUSION

In this paper, an ensemble feature selection method is proposed and tested. It is two-step process designed to obtain ensemble list. It involves creating the lists of features selected by feature selecton methods and applying the considered threshold on frequency and f-score of features in all selected features lists. At last, a classification method is applied to ensemble list features. Studying the results of Table 2 shows that combinations involving 5 better values in each evaluation metrics have the highest results in comparison to the results of basic feature selection methods and all features. Also, values of the best results in proposed method are higher in comparison to the results of web spam challenge Workshop(2008) and AUC method reported by Fdez-Glez et al.(2015). Moreover, values of the best results in proposed method are higher in comparison to the results of reported method by Keyhanipour & Moshiri (2013). Hence, the studies show that our method has better results. It is successful in terms of web spam detection.

In future, other classifiers can be applied to this method. In addition, in order to obtain the ensemble list of features, another method can be used instead of using frequency and F-score.

REFERENCES

Acharya, A., Cutts, M., Dean, J., Haahr, P., Henzinger, M., Hoelzle, U., . . . Tong, S. 2008. Information retrieval based on historical data: Google Patents.

Amitay, E., Carmel, D., Darlow, A., Lempel, R., & Soffer, A. Aug 2003. *The connectivity sonar: Detecting site functionality by structural patterns.* Paper presented at the the 14th ACM Conference on Hypertext and Hypermedia, Nottingham, UK. pp 38-47.

B, W., & B.D, D. 2006. Undue influence: eliminating the impact of link plagiarism on web search rankings. Paper presented at the Proceedings of the 2006 ACM symposium on Applied computing, Dijon, France. pp 1099-1104.

Becchetti, L., Castillo, C., Donato, D., Leonardi, S., & Baeza-Yates, R. 2006. *Using rank propagation and probabilistic counting for link-based spam detection.* Paper presented at the Proc. of WebKDD. pp 1-8.

Bellotti, T., Luo, Z., & Gammerman, A. 2006. Strangeness minimisation feature selection with confidence machines *Intelligent Data Engineering and Automated Learning–IDEAL 2006* (pp. 978-985): Springer.

Boldi, P. 2005. *TotalRank: Ranking without damping.* Paper presented at the Special interest tracks and posters of the 14th international conference on World Wide Web. pp 898-899.

Bolon-Canedo, V., Sanchez-Marono, N., & Alonso-Betanzos, A. 2011. Feature selection and classification in multiple class datasets: An application to KDD Cup 99 dataset. *Expert Systems with Applications, 38*(5): 5947-5957.

Cafarella, M., & Cutting, D. 2004. Building Nutch: Open Source Search. *Queue, 2*(2): 54-61. doi:10.1145/988392.988408

Chakrabarti, S., Joshi, M., & Tawde, V. 2001. Enhanced topic distillation using text, markup tags, and hyperlinks. Paper presented at the Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval, New Orleans, Louisiana, USA. pp 208-216.

Chen, Y.-W., & Lin, C.-J. 2006. Combining SVMs with various feature selection strategies *Feature extraction* (pp. 315-324): Springer.

Dahiya, S., Handa, S., & Singh, N. 2016. A rank aggregation algorithm for ensemble of multiple feature selection techniques in credit risk evaluation. *International Journal of Advanced Research in Artificial Intelligence, 5* 1-8.

Danandeh Oskouei, M., & Razavi, S. N. April 2015. Web Spam Detection Inspired by the Immune System. *International Journal of Computer Networks and Communications Security (IJCNCS), 3* (4). : page 191-199.

Davison, B. D. 2000. Recognizing nepotistic links on the web. *Artificial Intelligence for Web Search,* 23-28.

De Silva, A. M., & Leong, P. H. 2015. Feature selection *Grammar-Based Feature Generation for Time-Series Prediction* (13-24): Springer.

Dittman, D. J., Khoshgoftaar, T. M., Wald, R., & Napolitano, A. 2013. *Comparison of rank-based vs. score-based aggregation for ensemble gene selection.* Paper presented at the Information Reuse and Integration (IRI), 2013 IEEE 14th International Conference on. pp 225-231.

Eiron, N., McCurley, K. S., & Tomlin, J. A. 2004. Ranking the web frontier. Paper presented at the Proceedings of the 13th international conference on World Wide Web, New York, NY, USA. pp 309-318.

Fdez-Glez, J., Ruano-Ordas, D., Méndez, J. R., Fdez-Riverola, F., Laza, R., & Pavón, R. 2015. A dynamic model for integrating simple web spam classification techniques. *Expert Systems with Applications, 42*(21): 7969-7978.

Gao, K., Khoshgoftaar, T. M., & Napolitano, A. 2014. The Use of Ensemble-Based Data Preprocessing Techniques for Software Defect Prediction. *International Journal of Software Engineering and Knowledge Engineering, 24*(09): 1229-1253.

Guyon, I., & Elisseeff, A. 2003. An introduction to variable and feature selection. *The Journal of Machine Learning Research, 3* 1157-1182.

Gyongyi, Z., & Garcia-Molina, H. 2005. Web Spam Taxonomy. Paper presented at the First International Workshop on Adversarial Information Retrieval on the Web (AIRWeb 2005), Chiba, Japan. http://ilpubs.stanford.edu:8090/771/

Hall, M. A., & Smith, L. A. 1998. *Practical feature subset selection for machine learning.* Paper presented at Computer science’98 proceedings of the 21st Australasian computer science conference ACSC,pp 181-191.

Hall, M. A. 1998. Correlation-based feature subset selection for machine learning. Ph.D. thesis, University of Waikato, Hamilton, New Zealand.

Holte, R. C. 1993. Very simple classification rules perform well on most commonly used datasets. *Machine learning, 11*(1): 63-90.

Hoque, N., Singh, M., & Bhattacharyya, D. K. 2018. EFS-MI: an ensemble feature selection method for classification. *Complex & Intelligent Systems, 4*(2): 105-118.

Jayanthi, S., & Sasikala, S. 2010. Link Spam Detection Based on Dbspamclust with Fuzzy c-Means Clustering. *arXiv preprint arXiv:1101.0198*.

Jayanthi, S., & Sasikala, S. 2011. DBLC_SPAMCLUST: spamdexing detection by clustering clique-attacks in web search engines. *International Journal of Engineering Science and Technology, 3*(6).:page 4572-4580.

Jayanthi, S., & Sasikala, S. 2012. GAB_CLIQDET:-A diagnostics to Web Cancer (Web Link Spam) based on Genetic algorithm *Global Trends in Information Systems and Software Applications* (pp. 514-523): Springer Berlin Heidelberg.

Jayanthi, S., & Sasikala, S. 2013. Reptree Classifier For Identifying Link Spam In Web Search Engines. *Ictact Journal On Soft Computing, 3*(2):page 498-505.

Karimpour, J., Noroozi, A., & Alizadeh, S. 2012. Web Spam Detection by Learning from Small Labeled Samples. *International Journal of Computer Applications, 50*(21): 1-5. doi:10.5120/7924-0993

Keyhanipour, A. H., & Moshiri, B. 2013. *Designing a web spam classifier based on feature fusion in the Layered Multi-population Genetic Programming framework.* Paper presented at the Information Fusion (FUSION), 2013 16th International Conference on. pp 53-60.

Koc, L., Mazzuchi, T. A., & Sarkani, S. 2012. A network intrusion detection system based on a Hidden Naïve Bayes multiclass classifier. *Expert Systems with Applications, 39*(18): 13492-13500.

Kolari, P., Finin, T., & Joshi, A. 2006. *SVMs for the Blogosphere: Blog Identification and Splog Detection.* Paper presented at the AAAI Spring Symposium: Computational Approaches to Analyzing Weblogs. pp 92-99.

Kolari, P., Java, A., Finin, T., Oates, T., & Joshi, A. 2006. *Detecting spam blogs: A machine learning approach.* Paper presented at the Proceedings of the National Conference on Artificial Intelligence. pp 1351-1356.

Kononenko, I. 1994. *Estimating attributes: analysis and extensions of RELIEF.* Paper presented at the Machine Learning: ECML-94. pp 171-182.

Li, L., Shang, Y., & Zhang, W. 2002. Improvement of HITS-based algorithms on web documents. Paper presented at the Proceedings of the 11th international conference on World Wide Web, Honolulu, Hawaii, USA. pp 527-535.

Liu, H., Motoda, H., Setiono, R., & Zhao, Z. 2010. Feature Selection: An Ever Evolving Frontier in Data Mining. *Feature Selection in Data Mining (FSDM), 10:* 4-13.

Liu, H., & Setiono, R. 1995. *Chi2: Feature selection and discretization of numeric attributes.* Paper presented at the tai. pp 388-391.

Liu, H., & Setiono, R. 1996. *A probabilistic approach to feature selection-a filter solution.* Paper presented at the Proceedings of the Thirteenth International Conference on Machine Learning (ICML'96). pp 319-327.

Ng, A. Y., Zheng, A. X., & Jordan, M. I. 2001. *Stable algorithms for link analysis.* Paper presented at the Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval. pp 258-266.

Ntoulas, A., Najork, M., Manasse, M., & Fetterly, M. May 2006. *Detecting spam web pages through content analysis.* Paper presented at the the 15th International World Wide Web Conference, Edinburgh, Scotland.pp 83-92.

Osanaiye, O., Cai, H., Choo, K.-K. R., Dehghantanha, A., Xu, Z., & Dlodlo, M. 2016. Ensemble-based multi-filter feature selection method for DDoS detection in cloud computing. *EURASIP Journal on Wireless Communications and Networking*, *2016*(1): 130.

Prieto, V., Álvarez, M., López-García, R., & Cacheda, F. 2012. Analysis and Detection of Web Spam by Means of Web Content. In M. Salampasis & B. Larsen (Eds.), *Multidisciplinary Information Retrieval* (Vol. 7356, pp. 43-57): Springer Berlin Heidelberg.

Rungsawang, A., Taweesiriwate, A., & Manaskasemsak, B. 2011. Spam Host Detection Using Ant Colony Optimization. In J. J. Park, H. Arabnia, H.-B. Chang, & T. Shon (Eds.), *IT Convergence and Services* (Vol. 107, pp. 13-21): Springer Netherlands.

S.Sasikala, S. K. J. 2012. Genetic Algorithm and J48 Based Link Spamdexing Classifier for Web Search Engine. *International Journal of Computational Intelligence and Informatics*, *1*(4).: page 287-293.

Sahu, B., Dehuri, S., & Jagadev, A. K. 2017. An Ensemble Model using Genetic Algorithm for Feature Selection and rule mining using Apriori and FP-growth from Cancer Microarray data. *International Journal of Applied Engineering Research*, *12*(10): 2391-2408.

Seijo-Pardo, B., Porto-Díaz, I., Bolón-Canedo, V., & Alonso-Betanzos, A. 2017. Ensemble feature selection: homogeneous and heterogeneous approaches. *Knowledge-Based Systems, 118* 124-139.

Silva, R. M., Yamakami, A., & Alimeida, T. A. 2012. *An Analysis of Machine Learning Methods for Spam Host Detection.* Paper presented at the 11th International Conference on Machine Learning and Applications (ICMLA).pp 85-95.

Silwattananusarn, T., Kanarkard, W., & Tuamsuk, K. 2016. Enhanced classification accuracy for cardiotocogram data with ensemble feature selection and classifier ensemble. *Journal of Computer and Communications*, *4*(04): 20.

Wang, W., He, Y., Liu, J., & Gombault, S. 2015. Constructing important features from massive network traffic for lightweight intrusion detection. *IET Information Security, 9*(6): 374-379.

Westbrook, A., & Greene, R. 2002. *Using semantic analysis to classify search engine spam*. Class Project report at http://www. stanford. edu.

Workshop, O. W. o. t. W. S. C. 2008. Retrieved from http://Webspam.lip6.fr/wiki/pmwiki.php?n=Main.PhaseIII [Accessed 18 January 2008]

Zhang, L., Zhang, Y., Zhang, Y., & Li, X. 2006. *Exploring both content and link quality for anti-spamming.* Paper presented at the Computer and Information Technology, 2006. CIT'06. The Sixth IEEE International Conference on. pp. 37.

Mahdieh Danandeh Oskouei[1]
Seyed Naser Razavi[2]
[1]Department of Computer, Shabestar Branch, Islamic Azad University, Shabestar, Iran
[2]Department of Electrical and Computer Engineering, University of Tabriz, Iran
E-mail: [1]*mah.danandeh@gmail.com,* [2] *n.razavi@tabrizu.ac.ir*

Received: 11 June 2018
Accepted: 23 August 2018
Published: 27 December 2018

Appendix I

TABLE 1. Evaluation metrics of all features and feature selection methods on WEB SPAM-UK 2007 dataset.

		Precision	F-Measure	AUC	Accuracy
1	CfsSubsetEval+BestFirst	0.284	0.34	0.835	0.916
2	ChiSquaredAttributeEal+ Ranker	0.188	0.294	0.839	0.834
3	GainRatioAttributeEval+Ranker	0.27	0.34	0.837	0.909
4	OneRAttributeEval+Ranker	0.167	0.253	0.816	0.842
5	InfoGainAttributeEval+Ranker	0.183	0.288	0.839	0.829
6	ConsistencySubsetEval+Genetic algorithm	0.185	0.291	0.841	0.831
7	ReliefFAttributeEval+Ranker	0.19	0.276	0.801	0.864
8	F_score	0.219	0.312	0.841	0.878
10	All features	0.171	0.272	0.217	0.807

TABLE 2. Evaluation metrics of the proposed methods on WEB SPAM-UK 2007 dataset.

		Precision	F-Measure	AUC	Accuracy	Number of features
1	Combination 2,6	0.219	0.312	0.841	0.878	29
2	Combination 5,6	0.219	0.312	0.841	0.878	29
3	Combination 3,6	0.235	0.328	0.839	0.887	23
4	Combination 1,3	0.284	0.34	0.83	0.916	15
5	Combination 1,4	0.211	0.299	0.841	0.878	26
6	Combination 1,2,3	0.308	0.35	0.836	0.923	13
7	Combination 1,3,5	0.305	0.355	0.831	0.921	14
8	Combination 1,4,5	0.237	0.33	0.844	0.887	22
9	Combination 4,5,6	0.219	0.306	0.846	0.882	23
10	Combination 5,6,7	0.221	0.312	0.831	0.881	26
11	Combination 1,2,3,4	**0.328**	0.341	0.844	**0.93**	10
12	Combination 1,2,4,5	0.239	0.332	0.844	0.888	21
13	Combination 1,3,4,7	0.267	0.299	0.836	0.919	9
14	Combination 2,3,4,6	0.289	**0.359**	**0.851**	0.913	15
15	Combination 3,4,5,6	0.281	0.348	0.847	0.913	17
16	Combination 3,4,5,6,7	0.265	0.335	0.835	0.907	14

Appendix II

List of content features in WEBSPAMUK2007 Dataset.

HST_1
Number of words in the page (home page = hp)
HST_2
Number of words in the title (hp)
HST_3
Average word length (hp)
HST_4
Fraction of anchor text (hp)
HST_5
Fraction of visible text (hp)
HST_6
Compression rate of the hp
HST_7
Top 100 corpus precision (hp)
HST_8
Top 200 corpus precision (hp)
HST_9
Top 500 corpus precision (hp)
HST_10
Top 1000 corpus precision (hp)
HST_11
Top 100 corpus recall (hp)
HST_12
Top 200 corpus recall (hp)
HST_13
Top 500 corpus recall (hp)
HST_14
Top 1000 corpus recall (hp)
HST_15
Top 100 queries precision (hp)
HST_16
Top 200 queries precision (hp)
HST_17
Top 500 queries precision (hp)
HST_18
Top 1000 queries precision (hp)
HST_19
Top 100 queries recall (hp)
HST_20
Top 200 queries recall (hp)
HST_21
Top 500 queries recall (hp)
HST_22
Top 1000 queries recall (hp)
HST_23
Entropy (hp)
HST_24
Independent LH (hp)
HMG_25
Number of words in the page (page with max PageRank in the host = mp)
HMG_26
Number of words in the title (mp)
HMG_27
Average word length (mp)
HMG_28
Fraction of anchor text (mp)
HMG_29
Fraction of visible text (mp)
HMG_30
Compression rate (mp)
HMG_31
Top 100 corpus precision (mp)
HMG_32
Top 200 corpus precision (mp)
HMG_33
Top 500 corpus precision (mp)
HMG_34
Top 1000 corpus precision (mp)
HMG_35
Top 100 corpus recall (mp)
HMG_36
Top 200 corpus recall (mp)
HMG_37
Top 500 corpus recall (mp)
HMG_38
Top 1000 corpus recall (mp)
HMG_39
Top 100 queries precision (mp)
HMG_40
Top 200 queries precision (mp)
HMG_41
Top 500 queries precision (mp)
HMG_42
Top 1000 queries precision (mp)
HMG_43
Top 100 queries recall (mp)
HMG_44
Top 200 queries recall (mp)
HMG_45
Top 500 queries recall (mp)
HMG_46
Top 1000 queries recall (mp)
HMG_47
Entropy (mp)
HMG_48
Independent LH (mp)
AVG_49
Number of words in the page (average value for all pages in the host)
AVG_50
Number of words in the title (average value for all pages in the host)
AVG_51
Average word length (average value for all pages in the host)
AVG_52
Fraction of anchor text (average value for all pages in the host)
AVG_53
Fraction of visible text (average value for all pages in the host)
AVG_54
Compression rate (average value for all pages in the host)
AVG_55

Top 100 corpus precision (average value for all pages in the host)
AVG_56
Top 200 corpus precision (average value for all pages in the host)

AVG_57
Top 500 corpus precision (average value for all pages in the host)
AVG_58
Top 1000 corpus precision (average value for all pages in the host)
AVG_59
Top 100 corpus recall (average value for all pages in the host)
AVG_60
Top 200 corpus recall (average value for all pages in the host)
AVG_61
Top 500 corpus recall (average value for all pages in the host)
AVG_62
Top 1000 corpus recall (average value for all pages in the host)
AVG_63
Top 100 queries precision (average value for all pages in the host)
AVG_64
Top 200 queries precision (average value for all pages in the host)
AVG_65
Top 500 queries precision (average value for all pages in the host)
AVG_66
Top 1000 queries precision (average value for all pages in the host)
AVG_67
Top 100 queries recall (average value for all pages in the host)
AVG_68
Top 200 queries recall (average value for all pages in the host)
AVG_69
Top 500 queries recall (average value for all pages in the host)
AVG_70
Top 1000 queries recall (average value for all pages in the host)
AVG_71
Entropy (average value for all pages in the host)
AVG_72
Independent LH (average value for all pages in the host)
STD_73
Number of words in the page (Standard deviation for all pages in the host)
STD_74
Number of words in the title (Standard deviation for all pages in the host)
STD_75
Average word length (Standard deviation for all pages in the host)
STD_76
Fraction of anchor text (Standard deviation for all pages in the host)
STD_77
Fraction of visible text (Standard deviation for all pages in the host)
STD_78
Compression rate in the home page (Standard deviation for all pages in the host)
STD_79
Top 100 corpus precision (Standard deviation for all pages in the host)
STD_80
Top 200 corpus precision (Standard deviation for all pages in the host)
STD_81
Top 500 corpus precision (Standard deviation for all pages in the host)
STD_82
Top 1000 corpus precision (Standard deviation for all pages in the host)
STD_83
Top 100 corpus recall (Standard deviation for all pages in
the host)
STD_84
Top 200 corpus recall (Standard deviation for all pages in
the host)
STD_85
Top 500 corpus recall (Standard deviation for all pages in
the host)
STD_86
Top 1000 corpus recall (Standard deviation for all pages in the host)
STD_87
Top 100 queries precision (Standard deviation for all pages in the host)
STD_88
Top 200 queries precision (Standard deviation for all pages in the host)
STD_89
Top 500 queries precision (Standard deviation for all pages in the host)
STD_90
Top 1000 queries precision (Standard deviation for all pages in the host)
STD_91
Top 100 queries recall (Standard deviation for all pages in the host)
STD_92
Top 200 queries recall (Standard deviation for all pages in the host)
STD_93
Top 500 queries recall (Standard deviation for all pages in the host)
STD_94
Top 1000 queries recall (Standard deviation for all pages in the host)
STD_95
Entropy (Standard deviation for all pages in the host)
STD_96
Independent LH (Standard deviation for all pages in the host)